Channel Sharing of Competing Flows in Ad Hoc Networks *

Yu Wang J. J. Garcia-Luna-Aceves
Department of Computer Engineering
University of California
Santa Cruz, CA 95064, U.S.A.
Email: {ywang, jj}@cse.ucsc.edu

Abstract

This paper studies the fairness with which competing flows share the channel in ad hoc networks using collision avoidance protocols. It is shown that the required multi-hop coordination makes the backoff-based distributed fair queueing schemes less effective. Using extensive simulations of two competing flows with different underlying network configurations, it is shown that the commonly used flow contention graph is insufficient to model the contention among nodes and that various degrees of unfairness can take place. The fairness problem is more severe in TCP-based flows due to the required acknowledgment traffic, and TCP throughput is also negatively affected. A measurement-based fair scheme is analyzed in which nodes estimate their fair share of the channel from overheard traffic and adjust their backoff window accordingly (voluntarily); it is shown that such a scheme achieves much better fairness but sacrifices too much throughput. These results indicate that more explicit information exchange among contending nodes is mandatory to solve the fairness problem conclusively while maintaining reasonable throughput.

1 Introduction

Multi-hop ad hoc networks have received increasing interest in recent years, because they may be used in a large varieties of applications without the aid of any pre-existing infrastructure. Due to the scarcity of available wireless bandwidth, the design of efficient and effective medium access control (MAC) protocols that regulate nodes' access to a shared channel is of paramount importance. To mitigate the detrimental effects of hidden terminals [10], various MAC protocols have been proposed to avoid collisions and hence to enhance throughput. The most popular collision avoidance scheme to date includes a four-way handshake, i.e., RTS-CTS-data-Ack exchange between a pair of sending and receiving nodes. When collisions do occur, nodes have to back off random amounts of time and then attempt their channel access again. Due to its stability and long-term fairness, the binary exponential backoff (BEB) is favored in most MAC protocols and notably is adopted in the IEEE 802.11 MAC protocol [4], which is intended for wireless LANs and has been used extensively as the underlying MAC protocol in many routing protocols proposed for ad hoc networks. The BEB scheme is very effective in collision resolution when nodes face the same or similar level of contention. This is because when collisions occur, nodes double their contention window[1] and back off accordingly which reduces the contention significantly.

However, there are short-term and medium-term fairness problems associated with this scheme, because a node resets its contention window to the minimum size when it succeeds in sending a data packet. Accordingly, the node that last succeeds is much more aggressive by comparison in its next access to the channel and may monopolize the channel for a long time while other nodes suffer starvation. In fact, this problem in multi-hop networks was first addressed by Bharghavan et al. [2]. Some schemes have been proposed to alleviate the fairness problem. These schemes can be roughly divided into two categories. In the first category the goal is to achieve max-min fairness [1, 3, 9]. To be specific, these schemes try to reduce the ratio between maximum throughput and minimum throughput of flows, at either node's level or link's level. In the second category, the approach used in fair queueing for wireline networks is adapted to multi-hop ad hoc networks, taking into account the salient characteristics of such networks such as location-dependent contention, distributed coordination and possible spatial reuse [5–8, 11]. In these schemes, the contention among nodes is necessarily abstracted into a flow

*This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) under Grant No. DAAD19-01-C-0026 and by the US Air Force/OSR under Grant No. F49620-00-1-0330.

[1] Backoff timer is chosen from a uniform distribution that is bounded between 0 and the current contention window.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **2003**	2. REPORT TYPE	3. DATES COVERED **00-00-2003 to 00-00-2003**
4. TITLE AND SUBTITLE **Channel Sharing of Competing Flows in Ad Hoc Networks**		5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of California at Santa Cruz,Department of Computer Engineering,Santa Cruz,CA,95064**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**		
13. SUPPLEMENTARY NOTES		
14. ABSTRACT		
15. SUBJECT TERMS		

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**8**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

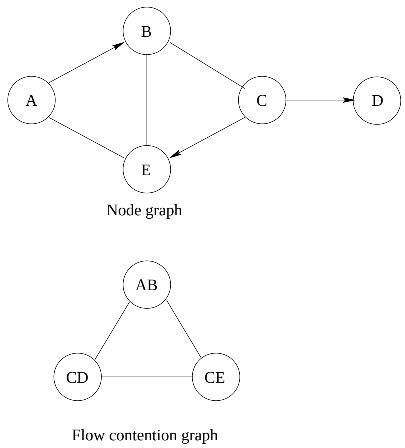

Figure 1. A simple network: node graph and flow contention graph

contention graph. Figure 1 shows an example of how this is done. Any two flows with adjacent vertices in the flow contention graph should not be scheduled to transmit at the same time. Each node decides its backoff time from the service tag (or start tag, depending on which fair queueing discipline for wireline networks is used for approximation) of their own flows and other flows in their local neighborhood or the whole network. Usually, the flow with the earliest service tag (or start tag) needs to back off for the minimum time so that its head-of-line (HOL) packet may be transmitted first.

The approaches in the second category are appealing because some service assurance may be provided for ad hoc networks if they can approximate the fair queueing algorithms used in wireline networks. In this paper, we address two problems associated with these schemes. The first is the coordination problem that results from the multi-hop characteristic of ad hoc networks. The difficulty of multi-hop coordination can make these backoff-based distributed fair queueing schemes less effective, which we discuss in Section 2. The second problem is that the flow contention graph is not sufficient to model the contention among nodes. By investigating how two competing flows share the available channel bandwidth with different underlying network topologies but with the same contention graph, we show that various degrees of fairness problems can take place. In Section 3, we describe a measurement-based fair scheme [1] in the first category. The scheme is simple and does not involve the overhead of explicit information exchange. Section 4 presents the simulation settings for two competing flows, in which we show that more than 10 different network topologies can have the same flow contention graph even though there are only two flows in these networks. Section 5 presents the results for TCP traffic, UDP traffic as well as a mix of TCP and UDP traffic in these networks. It is shown that the fairness problem does exist in the IEEE 802.11 MAC protocol and TCP-based flows suffer more, because the acknowledgment traffic from TCP can have negative effects on both fairness and throughput. It is also shown that, despite its simplicity, the measurement-based fair scheme achieves fairness at the price of too much degradation in throughput, especially for TCP-based flows. Supported by these results, Section 6 proposes that more explicit information exchange among contending nodes is mandatory and concludes this paper.

2 Multi-Hop Coordination

Most existing distributed fair queueing algorithms proposed in the literature for ad hoc networks modify the backoff scheme such that the sender of a flow with the minimum service tag (or start tag) within a contention region can transmit RTS almost the first by backing off for the minimum amount of time. Here the contention region includes all the flows that may collide with the interested flow. However, in the case of multi-hop networks, the node with the minimum backoff time cannot guarantee that its request for transmission with an RTS packet can succeed due to hidden terminals. This can happen when the difference between nodes' backoff time (measured in time slots for ease of discussion) is not large enough. In that case, another sender of a flow with the second-to-minimum service tag may also transmit an RTS, which leads to collisions.[2] This can be best illustrated by the network depicted in Figure 1. Here we consider only two flows, AB and CE. Suppose that node A has the minimum service tag and node C has the second-to-minimum service tag, such that the difference in backoff slots of A and C is b. Suppose that both RTS and CTS last n slots and ignore the propagation delay. If b is between 1 and $n-1$ (which is in fact $rts-1$), then C's RTS will collide with A's RTS at B. In this case, the difference between the two flows is not large enough for the one with the earlier service tag to access the channel successfully. Even worse, a node with the second-to-minimum service tag may even win its access to the channel over a node with the minimum service tag. For example, consider two flows AB and CD shown in Figure 1. If flow AB has the minimum service tag for its HOL packet and the backoff difference between the two flows is less than $n-1$, then before CTS from node B arrives at C, C can send an RTS packet, which makes D reply with CTS and B backoff and then C can transmit its HOL packet successfully to D, even though it does not have the minimum service tag. All nodes implement the protocol faithfully but fail to achieve the de-

[2] The node transmits because each node maintains only flow contention information and does not necessarily know the underlying network topology.

sired goal. In fact, this is referred to as *priority reversal* problem by Yang and Vaidya [13] though it is discussed in a different context.

Another not so severe case is that sometimes the flow with the second-to-minimum service tag may be penalized in some cases due to the flow with the minimum service tag. For example, consider flows AB and DC in Figure 1. Suppose that flow AB has the minimum service tag and DC has the second-to-minimum service tag. If the difference b is larger than $rts + 1$ and less than $rts + cts + data + ack + 1$, which is a large number, it is impossible for node D to initiate a successful handshake before nodes A and B finish theirs. Hence for any node like D, its transmission is almost doomed to failure even though it has backed off for such a long time. In addition, if the flow AB cannot finish sending its HOL packet in due time and the difference b between flow AB and flow DC drops below $rts + cts - 1$, then repeating collisions may occur if precautions are not taken.

It is evident from the above discussion that the required coordination among multi-hop nodes makes a backoff scheme derived from rankings of service tags less effective than what is expected, especially when only a flow contention graph is maintained and used in each node.

3 Measurement-based Fair Scheme

In this section, we describe the measurement-based fair scheme [12], which we then compare with the IEEE 802.11 MAC protocol using simulations. The rationale behind the scheme is surprisingly simple. Whenever a node sends or receives a packet, it updates its own estimation of its share (W_{ei}) or other nodes' share (W_{eo}) of the channel depending on the purpose of the packet. To avoid any explicit information exchange among these nodes, each node just treats all the nodes around itself as a single entity which competes against itself. For example, if a node sends an RTS packet, it will update W_{ei} because the RTS packet serves to reserve the channel for itself. If the node receives an RTS packet to itself, it will update W_{eo} because the RTS packet serves for other nodes. Details on the updating of W_{ei} and W_{eo} can be found in [12] and are not repeated here. Then the ratio between W_{ei} and W_{eo}, which is denoted by FI_e, serves as a fairness index to show whether a node is leading or lagging in channel access. If FI_e for a node is larger than a predefined constant C ($C > 1$), which implies that the node has obtained more than its fair share, then the node doubles its contention window (CW) from which backoff timer is derived. If FI_e lies between $1/C$ and C, then the node just holds on to its current CW as it estimates that both its neighbors and itself have obtained roughly equal fair share. If FI_e is smaller than $1/C$, then the node cuts its CW to a half to contend more vigorously for the channel. It should be noted that CW is bounded by the minimum and maximum values stipulated in the IEEE 802.11 standard. The measurement-based fair scheme is shown to be quite effective in the configurations investigated in [12] by sacrificing some throughput for better fairness.

However, this scheme may encounter the problem of severe throughput degradation in some cases, e.g., when two neighboring nodes are engaged in TCP-based connections. This can be explained as follows. In the measurement-based scheme, a node at the one end of a TCP connection continuously estimates its share and other node's share of the channel including the node on the other side of the connection. When this node sends one or a few data packets, it estimates that its use of the channel has exceeded its fair share and will increase its contention window accordingly. The other node behaves similarly. In this way, both nodes may have contention windows that are larger than necessary and the throughput is degraded due to the increased time wasted in waiting. The degradation in throughput can also happen in UDP-based flows in which two nodes take turns in channel access according to their own measurements. However, this phenomenon can be more conspicuous in TCP-based connections, because flow control and congestion avoidance in TCP may also be activated and can further slow down the channel access activities.

4 Two Competing Flows

We are interested in how the channel bandwidth is divided among two competing flows under different spatial contention characteristics and traffic patterns (either TCP or UDP). We do not consider two competing flows that originate from the same node because a node can perfectly avoid sending packets of the two flows at the same time. Furthermore, this is a local (or inner-node) scheduling problem that can be handled by the node itself. All the possible configurations are shown in Figures 2 and 3, in which a dashed line means that two nodes can hear each other's transmissions and an arrow indicates an active flow between two nodes. Nodes without any line in-between are hidden from each other. It is surprising to observe that there are so many variations even for such a simple case. We need to investigate how the different underlying network topologies can lead to various degrees of fairness problem via simulations which are presented in next section.

5 Simulation Results and Discussions

Simulations with the network configurations shown in Figures 2 and 3 were conducted with GloMoSim 2.0 [14]. The raw channel bandwidth is 2Mbps and the undering MAC protocol is IEEE 802.11 with direct sequence spread spectrum (DSSS) physical layer parameters. Table 1 shows the detailed parameters used throughout the simulations.

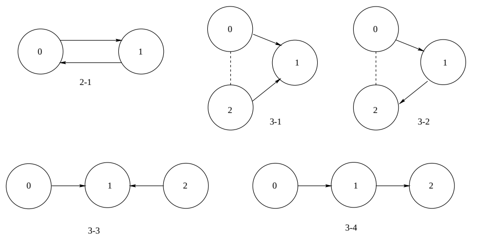

Figure 2. Networks with 2 or 3 nodes

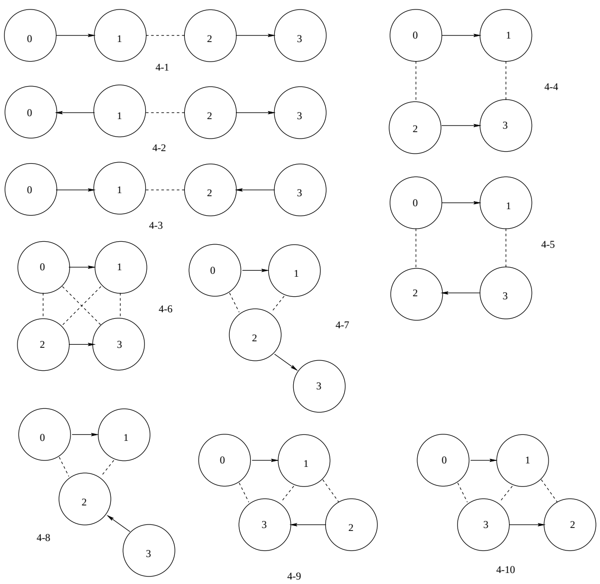

Figure 3. Networks with 4 nodes

RTS	CTS	data	ACK	DIFS	SIFS
20-byte	14-byte	1460-byte	14-byte	50μs	10μs

contention window	slot time	sync. time	prop. delay
31–1023	20μs	192μs	1μs

Table 1. IEEE 802.11 protocol configuration parameters

Section 5.1 presents simulations with the existing IEEE 802.11 MAC protocol under different traffic patterns, while Section 5.2 presents simulations with the measurement-based fair scheme and comparison of these two schemes.

5.1 The existing IEEE 802.11 MAC protocol

The simulations with the existing IEEE 802.11 MAC protocol include three sets. The first set shows the results for two competing TCP flows. We use the FTP/Generic application provided in GloMoSim, in which a client simply sends data packets to a server without the server sending any control information back to the client other than the acknowledgment packets required by TCP. Whenever a packet is indicated success of delivery by the transport layer (TCP), the client sends the next data packet. The second set shows the results for two competing UDP flows. We use the CBR application in which a client keeps sending data packets to a server at a constant bit rate, such that the sending queue is always non-empty. UDP is the underlying transport layer, thus no acknowledgment packets are sent back to the client. The third set shows the results for one FTP flow competing against one UDP flow, i.e., the FTP/Generic application vs. the CBR application. Except for the difference in the underlying transport layer, both flows generate packets of the same size and they are always backlogged.

We ran each configuration five times with different seed numbers and with a duration of 30 seconds, because we were interested in medium-term fairness in contrast to short-term or long-term fairness.[3] If the standard deviation of throughput is within 10% of the mean throughput, we show mean values only. Otherwise, we show both the mean and the standard deviation of throughput. In these cases, nodes take turns to monopolize the channel for a medium period of time, which we will discuss later.

The results for the three sets are shown in Tables 2–4. Due to space limitation, we omit insignificant results for some configurations. In Tables 2 and 3, the configurations with results that are worth noting are shown in bold face. If the two competing flows in these configurations are symmetric, they are also shown with asterisks.

From the results shown in Table 2, we have the following interesting observations for the two flows that use TCP as the underlying transport layer:

- The flow contention graph that has been used extensively in the past [5–8] is not enough to capture the characteristics of contending flows; we can observe radically different results even though all these two competing flows are the same in the flow contention graph.

- Even for two flows competing in a symmetric way, such as configuration 4-3, the channel bandwidth is not always divided evenly among these two flows. This is the medium-term fairness problem; otherwise, the BEB scheme used in IEEE 802.11 will still be able to achieve fairness in the long run due to the symmetry of the two competing flows.

  To illustrate the medium-term problem, we show a snapshot of one simulation run of configuration 4-3 in Figure 4. In this figure, the packet delivery times at MAC layer are recorded and shown. It is clear that the

[3]The simulation time is chosen elaborately to expose the medium-term fairness problem.

Table 2. Throughput comparison for TCP flows

Conf #	Flow 1	Flow 1 (kbps)	Flow 2	Flow 2 (kbps)	Aggregate (kbps)
3-1	$0 \rightarrow 1$	474	$2 \rightarrow 1$	468	942
3-4	$0 \rightarrow 1$	353	$1 \rightarrow 2$	547	899
4-1	$0 \rightarrow 1$	0	$2 \rightarrow 3$	918	926
4-2*	$1 \rightarrow 0$	515 ± 305	$2 \rightarrow 3$	419 ± 303	934
4-3*	$0 \rightarrow 1$	500 ± 407	$3 \rightarrow 2$	406 ± 424	906
4-4	$0 \rightarrow 1$	470 ± 78	$2 \rightarrow 3$	415 ± 70	884
4-5*	$0 \rightarrow 1$	498 ± 67	$3 \rightarrow 2$	372 ± 68	870
4-6	$0 \rightarrow 1$	475	$2 \rightarrow 3$	471	946
4-7	$0 \rightarrow 1$	924	$2 \rightarrow 3$	0	928
4-8	$0 \rightarrow 1$	926	$3 \rightarrow 2$	0	928
4-9	$0 \rightarrow 1$	427	$2 \rightarrow 3$	449	876
4-10*	$0 \rightarrow 1$	371	$3 \rightarrow 2$	529	901

Table 3. Throughput comparison for CBR flows

Conf #	Flow 1	Flow 1 (kbps)	Flow 2	Flow 2 (kbps)	Aggregate (kbps)
3-1	$0 \rightarrow 1$	806	$2 \rightarrow 1$	797	1600
3-3*	$0 \rightarrow 1$	761 ± 91	$2 \rightarrow 1$	782 ± 90	1540
3-4	$0 \rightarrow 1$	769	$1 \rightarrow 2$	839	1610
4-1	$0 \rightarrow 1$	83.4	$2 \rightarrow 3$	1500	1580
4-2	$1 \rightarrow 0$	820	$2 \rightarrow 3$	814	1630
4-3	$0 \rightarrow 1$	688	$3 \rightarrow 2$	709	1400
4-5	$0 \rightarrow 1$	725	$3 \rightarrow 2$	814	1540
4-7	$0 \rightarrow 1$	783	$2 \rightarrow 3$	824	1600
4-8	$0 \rightarrow 1$	1550	$3 \rightarrow 2$	28.2	1580
4-9*	$0 \rightarrow 1$	734 ± 98	$2 \rightarrow 3$	809 ± 94	1540
4-10	$0 \rightarrow 1$	781	$3 \rightarrow 2$	826	1610

Table 4. Throughput comparison for competing FTP and CBR flows

Conf #	Flow 1	Flow 1 (kbps)	Flow 2	Flow 2 (kbps)	Aggregate (kbps)
2-1	$0 \rightarrow 1$ (FTP)	0	$1 \rightarrow 0$ (CBR)	1570	1570
3-1	$0 \rightarrow 1$ (FTP)	355	$2 \rightarrow 1$ (CBR)	1000	1360
3-2a	$0 \rightarrow 1$ (FTP)	0	$1 \rightarrow 2$ (CBR)	1570	1570
3-2b	$0 \rightarrow 1$ (CBR)	991	$1 \rightarrow 2$ (FTP)	362	1360
3-3	$0 \rightarrow 1$ (FTP)	268 ± 58	$2 \rightarrow 1$ (CBR)	1110 ± 120	1370
3-4a	$0 \rightarrow 1$ (FTP)	0	$1 \rightarrow 0$ (CBR)	1570	1570
3-4b	$0 \rightarrow 1$ (CBR)	883	$1 \rightarrow 2$ (FTP)	427	1250
4-1a	$0 \rightarrow 1$ (FTP)	0	$2 \rightarrow 3$ (CBR)	1570	1570
4-1b	$0 \rightarrow 1$ (CBR)	102 ± 28	$2 \rightarrow 3$ (FTP)	865	967
4-2	$1 \rightarrow 0$ (FTP)	0	$2 \rightarrow 3$ (CBR)	1570	1570
4-3	$0 \rightarrow 1$ (FTP)	455 ± 93	$3 \rightarrow 2$ (CBR)	815 ± 133	1270
4-5	$0 \rightarrow 1$ (FTP)	297 ± 65	$3 \rightarrow 2$ (CBR)	1330	1370
4-7b	$0 \rightarrow 1$ (CBR)	1570	$2 \rightarrow 3$ (FTP)	0	1570
4-8a	$0 \rightarrow 1$ (FTP)	906	$3 \rightarrow 2$ (CBR)	412	948
4-8b	$0 \rightarrow 1$ (CBR)	1570	$1 \rightarrow 2$ (FTP)	0	1570
4-9	$0 \rightarrow 1$ (FTP)	311 ± 50	$2 \rightarrow 3$ (CBR)	1050 ± 110	1360
4-10b	$0 \rightarrow 1$ (CBR)	834 ± 94	$3 \rightarrow 2$ (TCP)	419 ± 55	1250

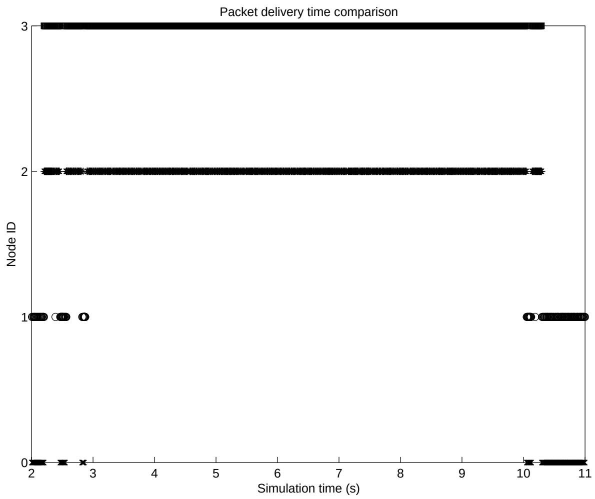

Figure 4. A snapshot of two competing TCP flows in conf. 4-3

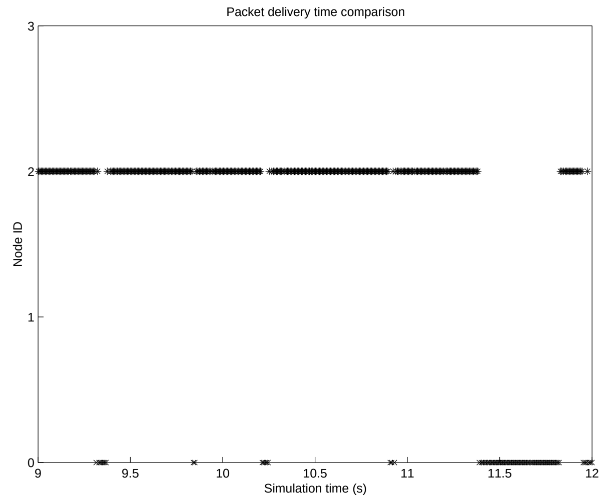

Figure 5. A snapshot of two competing UDP flows in conf. 4-9

TCP flow between nodes 0 and 1 may monopolize the channel for a very long time. In other simulation runs, it is also possible for the TCP flow between nodes 2 and 3 to monopolize the channel for a very long time.

- In some configurations, such as configurations 4-1, 4-7 and 4-8, the fairness problem is so severe that some TCP connections are in effect prevented from achieving any significant goodput. It should be noted that zero throughput does not mean that TCP connection is not set up. Instead, it is because of the extremely low throughput (on the order of a few kilobytes per second) for these flows that the statistics are not shown in these tables.

For UDP traffic, the fairness problem is not as severe as for TCP traffic. Serious fairness problems occur in only two configurations (configurations 4-1 and 4-8). There are two configurations (configurations 3-3 and 4-9) in which the fairness problem occurs but is not so severe. Some nodes have almost exclusive access to the shared channel for a certain amount of time that is not as long as in the case of TCP traffic. For example, a snapshot of one simulation run of configuration 4-9 is shown in Figure 5. it is clear that the flow from node 0 to 1 may experience very low throughput for several seconds.

When one FTP flow is competing against one CBR flow, as we can expect, the CBR flow achieves much higher throughput than the FTP flow in almost all the cases, except in configurations 4-1b and 4-8a, as shown in Table 4. This is because, the acknowledgment packets from the FTP server also have to fight their way back to the client in most cases and hence FTP throughput is greatly reduced. Exceptions to that behavior appear in configurations 4-1b and 4-8a, and may be briefly explained as follows. In configuration 4-1b, once node 2 sends a data packet successfully to node 3, node 3 initiates an RTS/CTS handshake for the acknowledgment packet. Because node 1 has already deferred its transmission for the previous successful handshake between nodes 2 and 3, it is very likely that node 0 is still in the backoff stage. Hence, node 2 receives the RTS from node 3 before node 1 receives node 0's RTS, even if node 0 starts sending its RTS during that time. Then, node 2 replies with a CTS, which is received successfully by node 3, while the same CTS may either collide with node 0's transmission at node 1 or be received by node 1. Whatever happens, the CTS from node 2 forces node 1 to defer its access again. In this way, the flow between nodes 2 and 3 is interrupted only sparsely by the flow from node 0 to node 1 and hence it can achieve much higher throughput than the other flow. The same line of reasoning explains the case for configuration 4-8.

It can also be observed that, in quite a few configurations, the TCP flow is prohibited from being set up or achieving any goodput. In such configurations, we have also experimented with some cases in which the start of CBR traffic is delayed purposely for some time after the start of FTP traffic. We find that the CBR traffic monopolizes the channel sooner or later and then TCP flow fails to achieve any goodput. It is clear that TCP traffic is at a significant disadvantage when competing against UDP traffic due to the acknowledgment packets required by TCP.

We also find that, the aggregate throughput of two competing flows is within 10% difference (e.g., configuration 4-5 vs. configuration 4-6) in Table 2 and 15% (e.g., configura-

Table 5. Throughput comparison for the IEEE 802.11 and the measurement-based fair scheme (MFS) – two CBR flows

Conf #	Scheme	Flow #	Throughput (kbps)	Flow #	Throughput (kbps)	Aggregate (kbps)
3-3	802.11	$0 \rightarrow 1$	761±91	$2 \rightarrow 1$	782±90	1540
	+MFS	$0 \rightarrow 1$	473	$2 \rightarrow 1$	471	944
4-1	802.11	$0 \rightarrow 1$	83.4	$2 \rightarrow 3$	1500	1580
	+MFS	$0 \rightarrow 1$	1010	$2 \rightarrow 3$	512	1520
4-8	802.11	$0 \rightarrow 1$	1550	$3 \rightarrow 2$	28.2	1580
	+MFS	$0 \rightarrow 1$	513	$3 \rightarrow 2$	1000	1520
4-9	802.11	$0 \rightarrow 1$	734±98	$2 \rightarrow 3$	809±94	1540
	+MFS	$0 \rightarrow 1$	473	$2 \rightarrow 3$	472	945

tion 4-3 vs. configuration 4-10) in Table 3 when traffic patterns are homogeneous despite the different configurations. This shows that aggregate throughput alone does not reveal the fairness problem easily. However, when traffic patterns are heterogeneous, the fairness problem is more prominent and has negative effects on the aggregate throughput, as shown in Table 4, where the difference may be more than 50% (e.g., configuration 4-8a vs. configuration 4-2).

5.2 The measurement-based fair scheme

As is already discussed in Section 3, the fairness problem is alleviated in the measurement-based fair scheme (MFS) by nodes' voluntary yielding of their access to the shared channel if they estimate that their use of the channel exceeds their deserved fair share. However, the throughput may degrade significantly due to the lack of explicit contention information exchanged among nodes, and hence nodes may end up backing off too long. The simulation results with the MFS are shown in Tables 5 and 6 and are compared against those with IEEE 802.11. For the sake of brevity, only the results for the configurations in which IEEE 802.11 MAC protocol exhibits the fairness problem are shown. It is clear that the MFS achieves far better fairness in almost all configurations shown here, but it sacrifices too much throughput. Hence despite its simplity, the MFS still calls for further improvement. In fact, it should make use of explicit information exchanged among nodes so that nodes can access and yield the channel more effectively and avoid unnecessary long waiting time.

6 Conclusion

In this paper, we have investigated the fairness problem in multi-hop ad hoc networks. We first pinpoint that the required multi-hop coordination can make less effective those distributed fair queueing schemes that depend on differentiated backoff to prioritize the access of a flow with the minimum service tag. Then we show that the commonly used flow contention graph is insufficient to model the contention among nodes via extensive simulations of two competing flows. Various degrees of the fairness problem can take place due to the different underlying network topologies, despite the same flow contention graph. Our simulation results also reveal that the reverse acknowledgment traffic required by TCP flows has negative effects on both throughput and fairness. On the one hand, TCP acknowledgment traffic can aggravate the fairness problem in the case of homogeneous traffic, because it may reinforce an already leading flow in some cases such that the flow can gain exclusive access to the shared channel for a long time. On the other hand, a TCP flow is at a disadvantage in competing against UDP flows, because TCP acknowledgment traffic has to fight its way back to the source, and throughput can be degraded in most cases due to the interference from unregulated UDP flows. We also show that schemes in which nodes voluntarily yield access to the shared channel according to their own measurement or estimation can suffer severe degradation in throughput despite their better fairness properties. In conclusion, we argue that the exchange of more explicit contention information among nodes and the effective use of such information are mandatory to solve the fairness problem conclusively, while maintaining reasonable throughput.

References

[1] B. Bensaou, Y. Wang, and C. C. Ko. Fair Medium Access in 802.11 Based Wireless Ad-Hoc Networks. In *IEEE/ACM MobiHoc Workshop*, Aug. 2000.

[2] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang. MACAW: A Media Access Protocol for Wireless LANs. In *Proc. of ACM SIGCOMM '94*, 1994.

[3] X. Huang and B. Bensaou. On Max-min Fairness and Scheduling in Wireless Ad-Hoc Networks: Analytical

Table 6. Throughput comparison for the IEEE 802.11 and the measurement-based fair scheme (MFS) – two FTP flows

Conf #	Scheme	Flow #	Throughput (kbps)	Flow #	Throughput (kbps)	Aggregate (kbps)
3-4	802.11	$0 \rightarrow 1$	353	$1 \rightarrow 2$	547	899
	+MFS	$0 \rightarrow 1$	137	$1 \rightarrow 2$	185	322
4-1	802.11	$0 \rightarrow 1$	0	$2 \rightarrow 3$	918	926
	+MFS	$0 \rightarrow 1$	214	$2 \rightarrow 3$	232	446
4-2	802.11	$1 \rightarrow 0$	515 ± 305	$2 \rightarrow 3$	419 ± 303	934
	+MFS	$1 \rightarrow 0$	250	$2 \rightarrow 3$	260	510
4-3	802.11	$0 \rightarrow 1$	500 ± 407	$3 \rightarrow 2$	406 ± 424	906
	+MFS	$0 \rightarrow 1$	219	$3 \rightarrow 2$	219	438
4-5	802.11	$0 \rightarrow 1$	498 ± 67	$3 \rightarrow 2$	372 ± 68	870
	+MFS	$0 \rightarrow 1$	305	$3 \rightarrow 2$	320	624
4-10	802.11	$0 \rightarrow 1$	371	$3 \rightarrow 2$	529	901
	+MFS	$0 \rightarrow 1$	223	$3 \rightarrow 2$	567	790

Framework and Implementation. In *ACM MobiHoc '01*, 2001.

[4] IEEE Computer Society LAN MAN Standards Committee, editor. *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Std 802.11-1997. The Institute of Electrical and Electronics Engineers, New York, 1997.

[5] H. Luo and S. Lu. A Topology-Independent Fair Queueing Model in Ad Hoc Wireless Networks. In *IEEE ICNP 2000*, Osaka, Japan, Nov. 2000.

[6] H. Luo, S. Lu, and V. Bharghavan. A New Model for Packet Scheduling in Multihop Wireless Networks. In *ACM Mobicom 2000*, Boston, MA, U.S., Aug. 2000.

[7] H. Luo, P. Medvedev, J. Cheng, and S. Lu. A Self-Coordinating Approach to Distributed Fair Queueing in Ad Hoc Wireless Networks. In *IEEE INFOCOM 2001*, Apr. 2001.

[8] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan. Achieving MAC Layer Fairness in Wireless Packet Networks. In *ACM Mobicom 2000*, Boston, MA, U.S., Aug. 2000.

[9] T. Ozugur, M. Naghshineh, P. Kermani, C. M. Olsen, B. Rezvani, and J. A. Copeland. Balanced Media Access Methods for Wireless Networks. In *Proc. of ACM/IEEE MOBICOM '98*, pages 21–32, Oct. 1998.

[10] F. A. Tobagi and L. Kleinrock. Packet Switching in Radio Channels: Part II - the Hidden Terminal Problem in Carrier Sense Multiple-access Modes and the Busy-tone Solution. *IEEE Trans. on Comm.*, 23(12):1417–1433, 1975.

[11] N. H. Vaidya, P. Bahl, and S. Gupta. Distributed Fair Scheduling in a Wireless LAN. In *ACM Mobicom 2000*, Boston, MA, U.S., Aug. 2000.

[12] Y. Wang and B. Bensaou. Achieving Fairness in IEEE 802.11 DFWMAC with Variable Packet Length. In *Proc. of IEEE Globecom*, San Antonio, Texas, U.S.A., Nov. 2001.

[13] X. Yang and N. Vaidya. Priority Scheduling in Wireless Ad Hoc Networks. In *ACM MobiHoc '02*, June 2002.

[14] X. Zeng, R. Bagrodia, and M. Gerla. GloMoSim: a Library for Parallel Simulation of Large-scale Wireless Networks. In *Proc. of the 12th Workshop on Parallel and Distributed Simulations*, May 1998.